

# Strict bounding of quantities of interest in computations based on domain decomposition\*

Valentine Rey<sup>1</sup>, Pierre Gosselet<sup>1</sup>, Christian Rey<sup>1,2</sup>

(1) LMT-Cachan / ENS Cachan, CNRS, Université Paris-Saclay  
61, avenue du président Wilson, 94235 Cachan, France

`valentine.rey@lmt.ens-cachan.fr`

`gosselet@lmt.ens-cachan.fr`

`christian.rey@lmt.ens-cachan.fr`

(2) SAFRAN Tech,

1, rue Geneviève Aubé, 78772 Magny les Hameaux, France

`christian.rey@safran.fr`

February 11, 2015

## Abstract

This paper deals with bounding the error on the estimation of quantities of interest obtained by finite element and domain decomposition methods. The proposed bounds are written in order to separate the two errors involved in the resolution of reference and adjoint problems : on the one hand the discretization error due to the finite element method and on the other hand the algebraic error due to the use of the iterative solver. Beside practical considerations on the parallel computation of the bounds, it is shown that the interface conformity can be slightly relaxed so that local enrichment or refinement are possible in the subdomains bearing singularities or quantities of interest which simplifies the improvement of the estimation. Academic assessments are given on 2D static linear mechanic problems.

---

\*NOTICE: this is the author's version of a work that was accepted for publication in CMAME. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Computer methods in Applied Mechanics and Engineering, 2015, doi: 10.1016/j.cma.2015.01.009.

**Keywords:** Verification; non-overlapping domain decomposition methods; quantity of interest; local refinement.

## 1 Introduction

In order to certify structures by computations, two fundamental abilities must be gathered: (i) conducting large computations which can be achieved by using non-overlapping domain decomposition methods to exploit parallel hardware [10, 19, 9], (ii) warranting the quality of the results which implies to use verification techniques. The computed approximation is then completed by an estimation of its distance to the exact solution, that distance being either on a global norm [1, 16, 13, 20] or on a selection of engineering quantities of interest [31, 22, 2].

This paper is the sequel of contributions by the authors in order to associate domain decomposition methods (DDM) and verification techniques. In [28] it was shown how admissible fields (aka balanced residuals) could be naturally computed in parallel when solving one finite element system with classical DDM solvers like FETI or BDD. This tool was necessary to obtain guaranteed error bounds. Moreover the proposed method encapsulated classical sequential techniques [18, 30, 27, 33] that could be employed independently on the subdomains. In [34] a global upper bound of the error was separated in two: the first contribution was purely due to the iterative solver employed in DDM and it could be made as small as wanted by doing more iterations, whereas the second contribution depended on the discretization and it was quasi-constant during the iterations. This result enabled us to stop iterations when the overall quality of the approximation could not be improved, which was much earlier than classical criteria on the norm of the residual would have suggested.

The purpose of this paper is to extend these works to the handling of quantities of interest. First we show how estimation on quantities of interest can be conducted easily within DDM with a separation of the contribution of the solver and of the discretization. Second we demonstrate that all the wanted properties can be preserved with nested kinematics at the interface which simplifies the improvement of the estimation by local adaptation.

The question of separating the contribution of the solver and of the discretization, in goal-oriented error estimation, was addressed in several other studies with different approaches. In [25, 24], the quantity of interest was the mean value of the unknown field in a region of the structure, the coarse mesh of the structure was seen as a decomposition of the true refined mesh which allowed to take advantage of the properties of FETI algorithm. In [29], the authors proposed constant-free asymptotic upper and lower bounds but they did not connect the iteration and

discretization contributions. In [21], error indicators separating contributions were developed in the context of multigrids methods for Poisson and Stokes equations. They lead to a gain in terms of CPU time but the bounds were not guaranteed. Using the dual weighted residual method [2] for goal-oriented error estimation, a balance of discretization and iteration errors was proposed for eigenvalue problems in [32]. Finally, in [40], the authors proposed goal-oriented a posteriori error estimator for problems discretized with mixed finite elements and multiscale domain decomposition that separated the contributions of the discretization of the subdomains from the contribution of the mortar interface.

In this paper, we employ classical extractor techniques [26, 39, 22], which leads to the definition of an adjoint problem which is solved at the same time as the forward problem using a block Krylov algorithm [37]. Thanks to [28, 34] we compute guaranteed upper and lower bounds of the quantity of interest at each iteration with the separation of the two sources of error for each problem (forward and adjoint), which enables us to drive the iterative solver by the error on the quantity of interest.

Moreover, since we consider quantities of interest defined on small supports, the adjoint problems have very localized solution. A simple strategy to improve the quality of the estimation is then to better the resolution of the adjoint problem either with a locally refined mesh [5] or using the partition of unity method and handbook techniques [6]. We show that within the framework of domain decomposition methods, it is possible to enrich or refine the kinematic of selected subdomains without impairing the exactness of the bound nor the ease of computation even if a certain incompatibility appears at the interface.

The paper is organized as follows. In Section 2, we present the goal-oriented error estimation within domain decomposition framework; we propose bounds on quantities of interest that separate the contributions of the discretization and of the solver. In Section 3, we show that nested interfaces can be employed which simplifies the improvement of the quality of the bounds by local refinement. Assessments are presented in Section 4 and Section 5 concludes the paper.

## 2 Error estimation on quantity of interest in substructured context

### 2.1 Substructured formulation of the forward and adjoint problems

#### 2.1.1 Reference problems

Let  $\mathbb{R}^d$  represent the physical space ( $d = 2$  or  $3$ ). Let us consider the static equilibrium of a (polyhedral) structure which occupies the open domain  $\Omega \subset \mathbb{R}^d$  and which is subjected to given body force  $\underline{f}$  within  $\Omega$ , to given traction force  $\underline{g}$  on  $\partial_g \Omega$  and to given displacement field  $\underline{u}_d$  on the complementary part of the boundary  $\partial_u \Omega$  ( $\text{meas}(\partial_u \Omega) \neq 0$ ). We assume that the structure undergoes small perturbations and that the material is linear elastic, characterized by Hooke's elasticity tensor  $\mathbb{H}$ . Let  $\underline{u}$  be the unknown displacement field,  $\underline{\varepsilon}(\underline{u})$  the symmetric part of the gradient of  $\underline{u}$ ,  $\underline{\sigma}$  the Cauchy stress tensor. In order to evaluate (localized) quantities of interest, we consider an adjoint problem loaded by a (linear) extractor  $\tilde{l}$  with small support. Fields related to the adjoint problem will always be noted with a tilde (for instance adjoint displacement is  $\tilde{\underline{u}}$ ).

Let  $\omega \subset \Omega$  be an open subset of  $\Omega$ . We introduce three affine subspaces, one linear space and one positive form:

- Affine subspace of kinematic admissible fields (KA-fields)

$$\text{KA}(\omega) = \left\{ \underline{u} \in (\mathbb{H}^1(\omega))^d, \underline{u} = \underline{u}_d \text{ on } \partial\omega \cap \partial_u \Omega \right\} \quad (1)$$

- Associated linear subspace  $\text{KA}^0$ :

$$\widetilde{\text{KA}}(\omega) = \text{KA}^0(\omega) = \left\{ \underline{u} \in (\mathbb{H}^1(\omega))^d, \underline{u} = 0 \text{ on } \partial\omega \setminus \partial_g \Omega \right\} \quad (2)$$

- Affine subspace of statically admissible fields (SA-fields) for the forward problem:

$$\begin{aligned} \text{SA}(\omega) = \left\{ \underline{\tau} \in (\mathbb{L}^2(\omega))_{\text{sym}}^{d \times d}; \forall \underline{v} \in \text{KA}^0(\omega), \right. \\ \left. \int_{\omega} \underline{\tau} : \underline{\varepsilon}(\underline{v}) d\Omega = \int_{\omega} \underline{f} \cdot \underline{v} d\Omega + \int_{\partial_g \Omega \cap \partial\omega} \underline{g} \cdot \underline{v} dS = l(\underline{v}) \right\} \quad (3) \end{aligned}$$

We note  $l$  the linear form which gathers the loads of the forward problem.

- Affine subspace of statically admissible fields (SA-fields) for the adjoint problem:

$$\widetilde{\text{SA}}(\omega) = \left\{ \underline{\underline{\tau}} \in (\text{L}^2(\omega))_{\text{sym}}^{d \times d}; \quad \forall \underline{v} \in \text{KA}^0(\omega), \quad \int_{\omega} \underline{\underline{\tau}} : \underline{\underline{\varepsilon}}(\underline{v}) d\omega = \tilde{l}(\underline{v}) \right\} \quad (4)$$

- Error in constitutive equation

$$e_{CR\omega}(\underline{u}, \underline{\underline{\sigma}}) = \|\underline{\underline{\sigma}} - \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u})\|_{\mathbb{H}^{-1}, \omega} \quad (5)$$

$$\text{where } \|\underline{x}\|_{\mathbb{H}^{-1}, \omega} = \sqrt{\int_{\omega} (\underline{x} : \mathbb{H}^{-1} : \underline{x}) d\omega}$$

The forward and adjoint problem set on  $\Omega$  can be formulated as:

$$\text{Find } (\underline{u}_{ex}, \underline{\underline{\sigma}}_{ex}) \in \text{KA}(\Omega) \times \text{SA}(\Omega) \text{ such that } e_{CR\Omega}(\underline{u}_{ex}, \underline{\underline{\sigma}}_{ex}) = 0 \quad (6)$$

$$\text{Find } (\tilde{\underline{u}}_{ex}, \tilde{\underline{\underline{\sigma}}}_{ex}) \in \text{KA}^0(\Omega) \times \widetilde{\text{SA}}(\Omega) \text{ such that } e_{CR\Omega}(\tilde{\underline{u}}_{ex}, \tilde{\underline{\underline{\sigma}}}_{ex}) = 0 \quad (7)$$

The solution to these problems, named “exact” solutions, exist and are unique.

### 2.1.2 Substructured formulation

Let us consider a decomposition of domain  $\Omega$  in  $N_{sd}$  open subsets  $\Omega^{(s)}$  such that  $\Omega^{(s)} \cap \Omega^{(s')} = \emptyset$  for  $s \neq s'$  and  $\bar{\Omega} = \bigcup_s \bar{\Omega}^{(s)}$ . The interface between subdomains  $\Gamma^{(s,s')} = \bar{\Omega}^{(s)} \cap \bar{\Omega}^{(s')}$  is supposed to be regular enough for traces of locally admissible fields to be well defined.  $\underline{n}$  is the outer normal vector.

The mechanical problem on the substructured configuration writes :

$$\forall s \left\{ \begin{array}{l} \underline{u}^{(s)} \in \text{KA}(\Omega^{(s)}) \\ \underline{\underline{\sigma}}^{(s)} \in \text{SA}(\Omega^{(s)}) \\ e_{CR\Omega^{(s)}}(\underline{u}^{(s)}, \underline{\underline{\sigma}}^{(s)}) = 0 \end{array} \right. \quad \text{and } \forall (s, s') \left\{ \begin{array}{l} \text{tr}(\underline{u}^{(s)}) = \text{tr}(\underline{u}^{(s')}) \text{ on } \Gamma^{(s,s')} \\ \underline{\underline{\sigma}}^{(s)} \cdot \underline{n}^{(s)} + \underline{\underline{\sigma}}^{(s')} \cdot \underline{n}^{(s')} = \underline{0} \text{ on } \Gamma^{(s,s')} \end{array} \right. \quad (8)$$

We assume the same domain decomposition is used for the forward and adjoint problems. The adjoint problem satisfies the same type of substructured formulation: the fields  $\underline{u}$  (respectively  $\tilde{\underline{u}}$ ) and  $\underline{\underline{\sigma}}$  (resp.  $\tilde{\underline{\underline{\sigma}}}$ ) are globally admissible if  $(\underline{u}^{(s)}, \underline{\underline{\sigma}}^{(s)})$  (resp.  $(\tilde{\underline{u}}^{(s)}, \tilde{\underline{\underline{\sigma}}})$ ) are kinematically and statically admissible inside each  $\Omega^{(s)}$  and if the displacements are continuous and tractions balanced on interfaces. The set of fields  $\underline{u}$  (or  $\tilde{\underline{u}}$ ) defined on  $\Omega$  such that they are kinematically admissible inside each domain  $\Omega^{(s)}$  without interface continuity is a broken space which we note  $\text{KA}(\bigcup \Omega^{(s)})$  and  $\text{KA}^0(\bigcup \Omega^{(s)})$ . Similarly, we denote  $\text{SA}(\bigcup \Omega^{(s)})$  the broken space of statically admissible fields inside subdomains.

*Remark.* The error in constitutive relation is by essence a parallel quantity:

$$\forall \underline{u} \in \text{KA}(\bigcup \Omega^{(s)}), \forall \underline{\underline{\sigma}} \in \text{SA}(\bigcup \Omega^{(s)}), \quad e_{CR_\Omega}(\underline{u}, \underline{\underline{\sigma}}) = \sqrt{\sum_s e_{CR_{\Omega^{(s)}}}^2(\underline{u}^{(s)}, \underline{\underline{\sigma}}^{(s)})} \quad (9)$$

## 2.2 Finite element approximation for the forward and adjoint problems

Both forward and adjoint problems are solved with a finite element approximation. Note that the mesh used to solve the adjoint problem can be completely different from the mesh used for the forward problem. We detail the finite element approximation for the forward problem (the principle is the same for the adjoint problem).

### 2.2.1 Global form

We associate with the mesh of  $\Omega$  the finite-element subspace  $\text{KA}_H(\Omega)$ . The finite element approximation consists in searching:

$$\begin{aligned} \underline{u}_H &\in \text{KA}_H(\Omega), \quad \underline{\underline{\sigma}}_H = \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_H) \\ \int_\Omega \underline{\underline{\sigma}}_H : \underline{\underline{\varepsilon}}(\underline{v}_H) d\Omega &= \int_\Omega \underline{f} \cdot \underline{v}_H d\Omega + \int_{\partial_g \Omega} \underline{g} \cdot \underline{v}_H dS = l(\underline{v}_H), \quad \forall \underline{v}_H \in \text{KA}_H^0(\Omega) \end{aligned} \quad (10)$$

We introduce the matrix  $\underline{\underline{\varphi}}_H$  of shape functions which form a basis of  $\text{KA}_H(\Omega)$  and the vector of nodal unknowns  $\mathbf{u}$ . Therefore, we have  $\underline{u}_H = \underline{\underline{\varphi}}_H \mathbf{u}$  and obtain the following linear system:

$$\mathbf{K} \mathbf{u} = \mathbf{f} \quad (11)$$

where  $\mathbf{K}$  is the (symmetric positive definite) stiffness matrix and  $\mathbf{f}$  is the vector of generalized forces (which includes imposed displacement).

### 2.2.2 Substructured form

For both problems, we assume that the mesh of  $\Omega$  and the substructuring are conforming. This hypothesis implies that each element only belongs to one subdomain and nodes are matching on the interfaces. Each degree of freedom is either located inside a subdomain (Subscript i) or on its boundary (Subscript b).

Let  $\mathbf{t}^{(s)}$  be the discrete trace operator, so that  $\mathbf{u}_b^{(s)} = \mathbf{t}^{(s)} \mathbf{u}^{(s)}$ . Let us introduce the unknown nodal reaction on the interface  $\boldsymbol{\lambda}^{(s)}$ , the equilibrium of each subdomain writes:

$$\mathbf{K}^{(s)} \mathbf{u}^{(s)} = \mathbf{f}^{(s)} + \mathbf{t}^{(s)T} \boldsymbol{\lambda}^{(s)} \quad (12)$$

Let  $(\mathbf{A}^{(s)})$  and  $(\mathbf{B}^{(s)})$  be the primal and dual assembly operator so that the discrete counterpart of the interface admissibility equations is:

$$\begin{cases} \sum_s \mathbf{B}^{(s)} \mathbf{t}^{(s)} \mathbf{u}^{(s)} = 0 \\ \sum_s \mathbf{A}^{(s)} \boldsymbol{\lambda}^{(s)} = 0 \end{cases} \quad (13)$$

Note that in the case when Subdomain  $s$  has not enough Dirichlet boundary conditions then the reaction has to balance the subdomain with respect to rigid body motions. Let  $\mathbf{R}^{(s)}$  be a basis of  $\ker(\mathbf{K}^{(s)})$ , we have:

$$\mathbf{R}^{(s)T} (\mathbf{f}^{(s)} + \mathbf{t}^{(s)T} \boldsymbol{\lambda}^{(s)}) = 0 \quad (14)$$

### 2.2.3 Domain decomposition solvers

First, let us briefly present the two classical domain decomposition solvers BDD and FETI (see Algorithm 2 with for now  $\mathbf{M} = \mathbf{N} = \mathbf{I}$  for BDD and [34] for FETI, see also [10] and its bibliography for more details). In BDD solver, a unique interface displacement is introduced so that the continuity of displacement is automatically verified. Using the primal Schur complement, the problem can be rewritten with interface quantities. In FETI solver, a unique set of nodal forces on the interface is introduced so that the balance of forces is verified. The dual Schur complement allows to write the problem with interface quantities.

Then, the resolution of interface problem is based on iterative Krylov solver, namely a preconditioned conjugate gradient. The classical preconditioners make use of scaled assembling operators  $(\tilde{\mathbf{A}}^{(s)})$  and  $(\tilde{\mathbf{B}}^{(s)})$  solutions of equations [35, 12]:

$$\sum_s \mathbf{A}^{(s)} \tilde{\mathbf{A}}^{(s)T} = \mathbf{I} \quad \text{and} \quad \sum_s \mathbf{B}^{(s)} \tilde{\mathbf{B}}^{(s)T} = \mathbf{I} \quad (15)$$

During the algorithm, two types of mechanical problems are solved locally (on each subdomain): problems with Dirichlet conditions on the interface (subscript D) and problems with Neumann conditions on the interface (subscript N). They correspond to the following procedures:

$$\left| \begin{array}{l} (\boldsymbol{\lambda}_D^{(s)}, \mathbf{u}_D^{(s)}) = \text{Solve}_D(\mathbf{u}_b^{(s)}, \mathbf{f}^{(s)}): \\ \left\{ \begin{array}{l} \mathbf{K}^{(s)} \mathbf{u}_D^{(s)} = \mathbf{f}^{(s)} + \mathbf{t}^{(s)T} \boldsymbol{\lambda}_D^{(s)} \\ \mathbf{t}^{(s)} \mathbf{u}_D^{(s)} = \mathbf{u}_b^{(s)} \end{array} \right. \end{array} \right| \left| \begin{array}{l} (\mathbf{u}_N^{(s)}) = \text{Solve}_N(\boldsymbol{\lambda}_N^{(s)}, \mathbf{f}^{(s)}) \\ \left\{ \begin{array}{l} \mathbf{K}^{(s)} \mathbf{u}_N^{(s)} = \mathbf{f}^{(s)} + \mathbf{t}^{(s)T} \boldsymbol{\lambda}_N^{(s)} \\ \text{where } (\boldsymbol{\lambda}_N^{(s)})_s \text{ satisfy Eq. (14)} \end{array} \right. \end{array} \right|$$

When developing these methods, we get:

$$\begin{aligned} (\mathbf{u}_D^{(s)})_i &= \mathbf{K}_{ii}^{(s)-1} \left( \mathbf{f}_i^{(s)} - \mathbf{K}_{ib}^{(s)} \mathbf{u}_b^{(s)} \right) \quad \text{and} \quad (\mathbf{u}_D^{(s)})_b = \mathbf{u}_b^{(s)} \\ \boldsymbol{\lambda}_D^{(s)} &= \mathbf{S}^{(s)} \mathbf{u}_b^{(s)} - \mathbf{f}_b^{(s)} + \mathbf{K}_{bi}^{(s)} \mathbf{K}_{ii}^{(s)-1} \mathbf{f}_i^{(s)} \\ \mathbf{u}_N^{(s)} &= \mathbf{K}^{(s)+} \left( \mathbf{f}^{(s)} + \mathbf{t}^{(s)T} \boldsymbol{\lambda}_N^{(s)} \right) \end{aligned}$$

where  $\mathbf{S}^{(s)} = \left( \mathbf{K}_{bb}^{(s)} - \mathbf{K}_{bi}^{(s)} \mathbf{K}_{ii}^{(s)-1} \mathbf{K}_{ib}^{(s)} \right)$  is the well-known Schur complement matrix, and  $\mathbf{K}^{(s)+}$  is a pseudo-inverse of  $\mathbf{K}^{(s)}$ . Thanks to projector ( $\mathbf{P}_1$ ) and initialization (method `Initialize`), the Neumann problems are well-posed (see [10]).

If the forward and adjoint problems are approximated on the same mesh, they share the same stiffness matrix and they can be solved simultaneously using a block-conjugate gradient. In Algorithm 2, we use capital letter to note blocks of forward/adjoint vectors: typically  $\mathbf{U} = [\mathbf{u}, \tilde{\mathbf{u}}]$ , same notation holds for the residual  $\mathbf{R} = [\mathbf{r}, \tilde{\mathbf{r}}]$ , preconditioned residual  $\mathbf{Z} = [\mathbf{z}, \tilde{\mathbf{z}}]$ , search directions  $\mathbf{W}$  and  $\mathbf{Q}$ ;  $\alpha$  is a  $2 \times 2$  matrix. One has to pay attention to the potential (yet unlikely) singularity of the  $2 \times 2$  matrix  $(\mathbf{Q}^T \mathbf{W})$  which needs to be inverted.

#### 2.2.4 Description of admissible fields

At every iteration of the solver, a totally parallel computation of the following fields is possible [28]:

- $\underline{u}_D = (\underline{u}_D^{(s)})_s \in \text{KA}(\Omega)$  and  $\tilde{\underline{u}}_D = (\tilde{\underline{u}}_D^{(s)})_s \in \text{KA}^0(\Omega)$ : globally admissible displacement fields,  $(\boldsymbol{\lambda}_D^{(s)}, \tilde{\boldsymbol{\lambda}}_D^{(s)})$  are the associated nodal reaction which are not balanced before convergence.
- $\underline{u}_N = (\underline{u}_N^{(s)})_s \in \text{KA}(\bigcup \Omega^{(s)})$  and  $\tilde{\underline{u}}_N = (\tilde{\underline{u}}_N^{(s)})_s \in \text{KA}^0(\bigcup \Omega^{(s)})$ : displacements associated to a given balanced Neumann condition  $\boldsymbol{\lambda}_N^{(s)}$  (resp.  $\tilde{\boldsymbol{\lambda}}_N^{(s)}$ ), they are not necessarily continuous across interfaces.
- $\underline{\underline{\sigma}}_N^{(s)} = \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_N^{(s)})$ : the stress field associated to  $\underline{u}_N^{(s)}$ . It can be employed (with additional input  $\boldsymbol{\lambda}_N^{(s)}$ ) to build stress fields  $\hat{\underline{\underline{\sigma}}}_N^{(s)}$  which are statically admissible  $\hat{\underline{\underline{\sigma}}}_N = (\hat{\underline{\underline{\sigma}}}_N^{(s)})_s \in \text{SA}(\Omega)$ . Same property holds for the adjoint problem, we can build  $\hat{\tilde{\underline{\underline{\sigma}}}}_N = (\hat{\tilde{\underline{\underline{\sigma}}}}_N^{(s)})_s \in \widetilde{\text{SA}}(\Omega)$ .

As a consequence, for both forward and adjoint problems, kinematically displacements fields and statically stress fields are available even if the solver is not converged. Those fields are used to compute global error estimation for both problems.



Indeed, we have the following guaranteed upper bounds which can be computed in parallel [28]:

$$\begin{aligned}\|\underline{u}_{ex} - \underline{u}_D\|_\Omega &\leq e_{CR_\Omega}(\underline{u}_D, \hat{\underline{\sigma}}_N) \\ \|\tilde{\underline{u}}_{ex} - \tilde{\underline{u}}_D\|_\Omega &\leq e_{CR_\Omega}(\tilde{\underline{u}}_D, \hat{\tilde{\underline{\sigma}}}_N)\end{aligned}\tag{16}$$

where  $\|\underline{x}\|_\Omega = \|\underline{\underline{\varepsilon}}(\underline{x})\|_{\mathbb{H}^{-1}, \Omega}$ .

## 2.3 Error estimation of quantities of interest

As said earlier the adjoint problem is meant to extract one quantity of interest in the forward problem. Let  $I_{ex} = \tilde{l}(\underline{u}_{ex})$  be the unknown exact value of the quantity of interest.  $\tilde{l}(\underline{u}_D)$  is an approximation of this quantity of interest.

### 2.3.1 Upper and lower bounds with separated contributions

We demonstrate two results which provide upper and lower bounds on the quantity of interest, as a product of error estimates on the forward and adjoint problems with, for each of these factors, a separation of contributions of the solver and of the discretization.

**Property 1.** *Using notation of Algorithm 2, we have*

$$|I_{ex} - \tilde{l}(\underline{u}_D) - I_{HH,1}| \leq \left( \sqrt{\mathbf{r}^T \mathbf{z}} + e_{CR_\Omega}(\underline{u}_N, \hat{\underline{\sigma}}_N) \right) \left( \sqrt{\tilde{\mathbf{r}}^T \tilde{\mathbf{z}}} + e_{CR_\Omega}(\tilde{\underline{u}}_N, \hat{\tilde{\underline{\sigma}}}_N) \right) \tag{17}$$

with

$$I_{HH,1} = \int_\Omega (\hat{\underline{\sigma}}_N - \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_D)) : \underline{\underline{\varepsilon}}(\tilde{\underline{u}}_D) d\Omega \tag{18}$$

*Proof.* This result is a direct application of the results demonstrated in [14, 15]:

$$|I_{ex} - \tilde{l}(\underline{u}_D) - I_{HH,1}| \leq e_{CR_\Omega}(\underline{u}_D, \hat{\underline{\sigma}}_N) e_{CR_\Omega}(\tilde{\underline{u}}_D, \hat{\tilde{\underline{\sigma}}}_N) \tag{19}$$

with  $I_{HH,1}$  given in (18). The separation between the sources of error is obtained using the triangular inequality and Lemma 2 presented in [34] (which proves the equality  $\|\underline{u}_N - \underline{u}_D\|_\Omega = \sqrt{\mathbf{r}^T \mathbf{z}}$ ), it slightly improves the result given in remark 25 of [34]:

$$\begin{aligned}e_{CR_\Omega}(\underline{u}_D, \hat{\underline{\sigma}}_N) &= \|\hat{\underline{\sigma}}_N - \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_D)\|_{\mathbb{H}^{-1}, \Omega} \\ &= \|\hat{\underline{\sigma}}_N - \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_N) + \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_N - \underline{u}_D)\|_{\mathbb{H}^{-1}, \Omega} \\ &\leq \|\hat{\underline{\sigma}}_N - \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_N)\|_{\mathbb{H}^{-1}, \Omega} + \|\underline{u}_N - \underline{u}_D\|_\Omega \\ &\leq e_{CR_\Omega}(\underline{u}_N, \hat{\underline{\sigma}}_N) + \sqrt{\mathbf{r}^T \mathbf{z}}\end{aligned}\tag{20}$$

□

The quantity  $\sqrt{\mathbf{r}^T \mathbf{z}}$  (respectively  $\sqrt{\tilde{\mathbf{r}}^T \tilde{\mathbf{z}}}$ ) is a measure of the residual of the forward (resp. adjoint) problem: it corresponds to a pure algebraic error. The quantity  $e_{CR_{\Omega(s)}}(\underline{u}_N, \hat{\underline{\sigma}}_N)$  (resp.  $e_{CR_{\Omega(s)}}(\tilde{\underline{u}}_N, \hat{\tilde{\underline{\sigma}}}_N)$ ) is a parallel estimate of the discretization error of the forward (resp. adjoint) problem. As a consequence, we can define a new stopping criterion for the solver: the idea is to stop iterations when the algebraic error becomes negligible in comparison with the discretization error. This new stopping criterion is described in alg. 1.

---

**Algorithm 1:** DD block-solver with adapted stopping criterion

---

```

Set  $\alpha > 1$ ,  $\beta \geq 1$ ,  $\tilde{\alpha} > 1$  and  $\tilde{\beta} \geq 1$ ;
Initialize, get  $(\mathbf{U}_N^{(s)}, \mathbf{\Lambda}_N^{(s)}, \mathbf{R}, \mathbf{Z})$ ;
Estimate discretization error for reference problem  $e^2 = \sum_s e_{CR_{\Omega(s)}}^2(\underline{u}_N^{(s)}, \hat{\underline{\sigma}}_N^{(s)})$ 
and adjoint problem  $\tilde{e}^2 = \sum_s e_{CR_{\Omega(s)}}^2(\tilde{\underline{u}}_N^{(s)}, \hat{\tilde{\underline{\sigma}}}_N^{(s)})$ ;
while  $\sqrt{\mathbf{r}^T \mathbf{z}} > e/\alpha$  and  $\sqrt{\tilde{\mathbf{r}}^T \tilde{\mathbf{z}}} > \tilde{e}/\tilde{\alpha}$  do
    while  $\sqrt{\mathbf{r}^T \mathbf{z}} > e/(\alpha\beta)$  and  $\sqrt{\tilde{\mathbf{r}}^T \tilde{\mathbf{z}}} > \tilde{e}/\tilde{\alpha}\tilde{\beta}$  do
        | Make Alg. 2 iterations, get  $(\mathbf{U}_N^{(s)}, \mathbf{\Lambda}_N^{(s)})$ 
    end
    Update error estimators  $e^2 = \sum_s e_{CR_{\Omega(s)}}^2(\underline{u}_N^{(s)}, \hat{\underline{\sigma}}_N^{(s)})$  and
     $\tilde{e}^2 = \sum_s e_{CR_{\Omega(s)}}^2(\tilde{\underline{u}}_N^{(s)}, \hat{\tilde{\underline{\sigma}}}_N^{(s)})$ ;
end

```

---

The objective being to have the norm of the residual  $\alpha$  times smaller than the estimation of the discretization error for both problems ( $\alpha = \tilde{\alpha} = 10$  is a typical value), the coefficient  $\beta = \tilde{\beta}$  (typically 2) takes into account the small variation of the estimate  $e$  and  $\tilde{e}$  of the discretization error along iterations.

A second (twice better) bound can be derived in a similar way using another result of [14, 15].

**Property 2.**

$$|I_{ex} - \tilde{l}(\underline{u}_D) - I_{HH,2}| \leq \frac{1}{2}(\sqrt{\mathbf{r}^T \mathbf{z}} + e_{CR_{\Omega}}(\underline{u}_N, \hat{\underline{\sigma}}_N))(\sqrt{\tilde{\mathbf{r}}^T \tilde{\mathbf{z}}} + e_{CR_{\Omega}}(\tilde{\underline{u}}_N, \hat{\tilde{\underline{\sigma}}}_N)) \quad (21)$$

with

$$I_{HH,2} = \frac{1}{2} \int_{\Omega} \left( \hat{\underline{\sigma}}_N - \mathbb{H}_{\underline{\varepsilon}}(\underline{u}_D) \right) : \mathbb{H}^{-1} : \left( \hat{\tilde{\underline{\sigma}}}_N + \mathbb{H}_{\underline{\varepsilon}}(\tilde{\underline{u}}_D) \right) d\Omega \quad (22)$$

### 2.3.2 Properties of $I_{HH,1}$

If the forward and adjoint problems share the same mesh, the quantity  $I_{HH,1}$  can be rewritten in terms of quantity on the interface, which simplifies its evaluation:

**Property 3.**

$$I_{HH,1} = \sum_s \boldsymbol{\lambda}_D^{(s)T} \mathbf{t}^{(s)} \tilde{\mathbf{u}}_D^{(s)} \quad (23)$$

*Proof.* The forward problems verified by  $\underline{\mathbf{u}}_D \in \text{KA}(\Omega)$  and  $\underline{\mathbf{u}}_{ex} \in \text{KA}(\Omega)$  with a testing field  $\tilde{\underline{\mathbf{u}}}_D \in \text{KA}^0(\Omega)$  read:

$$\sum_s \int_{\Omega_s} \underline{\underline{\varepsilon}}(\underline{\mathbf{u}}_D^{(s)}) : \mathbb{H} : \underline{\underline{\varepsilon}}(\tilde{\underline{\mathbf{u}}}_D^{(s)}) d\Omega = \sum_s l^{(s)}(\tilde{\underline{\mathbf{u}}}_D^{(s)}) + \sum_s \boldsymbol{\lambda}_D^{(s)T} \mathbf{t}^{(s)} \tilde{\mathbf{u}}_D^{(s)} \quad (24)$$

$$\sum_s \int_{\Omega_s} \underline{\underline{\varepsilon}}(\underline{\mathbf{u}}_{ex}^{(s)}) : \mathbb{H} : \underline{\underline{\varepsilon}}(\tilde{\underline{\mathbf{u}}}_D^{(s)}) d\Omega = \sum_s l^{(s)}(\tilde{\underline{\mathbf{u}}}_D^{(s)}) + \sum_s \int_{\Gamma^{(s)}} (\underline{\underline{\sigma}}_{ex} \cdot \underline{\mathbf{n}}^{(s)}) \cdot \tilde{\underline{\mathbf{u}}}_D^{(s)} ds \quad (25)$$

Since  $\tilde{\underline{\mathbf{u}}}_D = (\tilde{\underline{\mathbf{u}}}_D^{(s)})_s \in \text{KA}^0(\Omega)$  and  $\underline{\mathbf{u}}_{ex}$  is the exact solution, we have:

$$\sum_s \int_{\Gamma^{(s)}} (\underline{\underline{\sigma}}_{ex} \cdot \underline{\mathbf{n}}^{(s)}) \cdot \tilde{\underline{\mathbf{u}}}_D^{(s)} ds = 0 \quad (26)$$

while since  $\hat{\underline{\underline{\varepsilon}}}_N$  is statically admissible and  $(\tilde{\underline{\mathbf{u}}}_D^{(s)})_s \in \text{KA}^0(\Omega)$ , we have:

$$I_{HH,1} = \int_{\Omega} (\hat{\underline{\underline{\varepsilon}}}_N - \mathbb{H}\underline{\underline{\varepsilon}}(\underline{\mathbf{u}}_D)) : \underline{\underline{\varepsilon}}(\tilde{\underline{\mathbf{u}}}_D) d\Omega = \int_{\Omega} (\underline{\underline{\sigma}}_{ex} - \mathbb{H}\underline{\underline{\varepsilon}}(\underline{\mathbf{u}}_D)) : \underline{\underline{\varepsilon}}(\tilde{\underline{\mathbf{u}}}_D) d\Omega \quad (27)$$

Therefore, by substracting equation (24) from equation (25), we obtain (23).  $\square$

When the solver has reached convergence, if the meshes of the forward and adjoint problems are identical, the term  $I_{HH,1}$  is equal to zero (as a consequence of the Galerkin orthogonality). Thanks to the new expression of  $I_{HH,1}$ , we can prove the following property.

**Property 4.** *Using a BDD solver, at each iteration,  $I_{HH,1} = 0$ .*

*Proof.* Using the notation of BDD's algorithm (alg. 2 with  $\mathbf{M} = \mathbf{N} = \mathbf{I}$ ), we have:

$$I_{HH,1} = \mathbf{r}^T \tilde{\mathbf{u}} \quad (28)$$

The iterations of the block-CG can be defined by the following Galerkin scheme [23, 36]:

$$\begin{cases} \mathbf{U}_i \in \mathbf{U}_0 + \mathcal{K}_i \\ \mathbf{R}_i \perp \mathcal{K}_i \end{cases} \quad (29)$$

where  $\mathbf{U} = [\mathbf{u}, \tilde{\mathbf{u}}]$ ,  $\mathbf{R} = [\mathbf{r}, \tilde{\mathbf{r}}]$ , and  $\mathcal{K}_i$  is the Krylov subspace associated with the preconditioned operator and the initial block of residuals. This property implies that

$$I_{HH,1} = \mathbf{r}^T \tilde{\mathbf{u}} = \mathbf{r}_0^T \tilde{\mathbf{u}}_0 \quad (30)$$

In order to prove the nullity of this term, one has to look more in detail at BDD's initialization. We refer to [10] and simply recall that  $\mathbf{r}_0 \in \text{range}(\mathbf{P}_1^T)$  while  $\tilde{\mathbf{u}}_0 \in \ker(\mathbf{P}_1)$ .  $\square$

## 2.4 Assessment

Let us consider a rectangular linear elastic structure  $\Omega = [-3l; 3l] \times [-3l; 3l]$  with homogeneous Dirichlet boundary conditions. Young modulus is 1 Pa and Poisson coefficient is 0.3. The domain is subjected to a polynomial body force such that the exact solution is known:

$$\underline{u}_{ex} = (x + 3l)(x - 3l)(y + 3l)(y - 3l) \left( (y - 3l)^2 \underline{e}_x + (y + 3l) \underline{e}_y \right)$$

The quantity of interest is the average of the normal stress in the direction  $\underline{e}_y$  in a small square of side  $l/10$  centered at the origin.

$$\tilde{l}(\underline{u}) = \frac{1}{\text{mes}(\omega)} \int_{\omega} (\underline{e}_y \otimes \underline{e}_y) : \underline{\underline{\varepsilon}}(\underline{u}) d\omega \quad (31)$$

The mesh is made out of first order Lagrange triangles, which leads to 2352 degrees of freedom. As shown in Figure 1 the structure is decomposed into 9 regular subdomains.

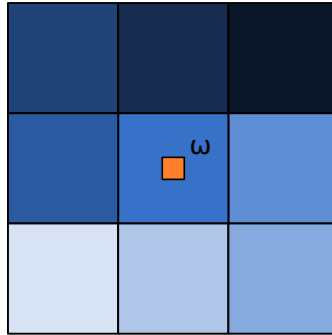


Figure 1: Substructuring

We solved the problem with a BDD solver (primal approach). We used the Element Equilibration Technique [17] to build statically admissible stress fields; each elementary problem was solved using an interpolation of degree 4. For the

sake of simplicity, we note:  $e = e_{CR_\Omega}(\underline{u}_N, \hat{\underline{\sigma}}_N)$  and  $\tilde{e} = e_{CR_\Omega}(\tilde{\underline{u}}_N, \hat{\tilde{\underline{\sigma}}}_N)$ . The results in Table 1 are given when the solver is converged (when the residual of both forward and adjoint problem are negligible with respect to the loads); they show that the error estimation in a substructured context is as efficient as in a sequential context. We recall that the slight difference is due to the fact that the static admissible fields are not identical in sequential and in domain decomposition methods.

case	$e$	$\tilde{e}$	$I_H$	$I_{HH,2}$	$\frac{1}{2}e\tilde{e}$	$I_{ex}$
Sequential	2.417	10.23	2.916	$3.305 \cdot 10^{-4}$	12.36	2.916
Substructured	2.418	10.23	2.916	$3.858 \cdot 10^{-4}$	12.37	2.916

Table 1: Substructured and sequential goal-oriented error estimations

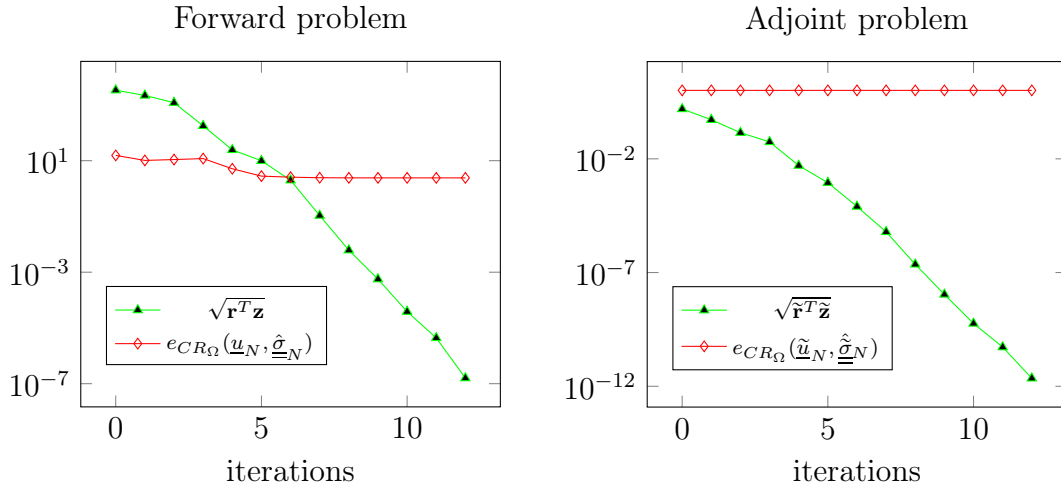


Figure 2: Error estimates with separation of sources

The plots on Figure 2 show the evolution of the two contributions to the bound during the iterations of the conjugate gradient. We observe that the discretization error is almost constant whereas the algebraic error decreases along the iterations. On that example after 6 iterations, the solver errors of both forward and adjoint problems are negligible with respect to the contribution of the discretization, extra iterations are then useless, they do not improve the quality of the approximation.

## 3 Improvement of the bound by local refinement

### 3.1 Motivation

The upper bounds obtained in Property 1 and Property 2 always involve the product between the error estimators of the forward and adjoint problem. Accordingly increasing the quality of either (or both) the forward or the adjoint problems necessarily improves the bounds. For instance, since the solution of the adjoint problem is highly localized, it is often easier to better its estimation, either by mesh refinement or by the injection of a priori knowledge [11, 41, 5, 6].

Adapting a mesh, even for a local quantity of interest, is a global problem: simple algorithms like the one based on bisection [38] propagate on the whole domain; parallel remeshing involves exchanges between neighbors (and iterations if good load balance is also wished) [8]. Enrichment also implies integrating functions on parts of the domain. In this section, we try to benefit the domain decomposition in order to simplify these procedure. The idea is to improve the approximation only on the subdomains which most contribute to the estimators. We show that a class of adaptation algorithms can be realized completely locally without impairing the exactness and the computability of the bound: algorithms which modify the boundary of the treated subdomains only by adding new degrees of freedom. This leads to what we call “nested interfaces” which means that the trace of the initial (coarse) kinematic is a subspace of the trace of the new (fine) kinematic.

Typically, bisections stopped at the boundary of the subdomains belongs to that class of algorithms, so do hierarchical h and p refinement and most enrichment methods. In the examples, for simplicity reasons, we use uniform h-refinement in the subdomains. Of course such procedures open the question of balancing the computational load between subdomains, which will be the subject of future work. Anyhow, a classical answer is to split the structure into many subdomains and to randomly attribute several subdomains to each computing unit with the hope that the to-be-refined subdomains will be evenly distributed.

The rest of the section is organized as follows. In Subsection 3.2 we show how a master/slave algorithm enables us to treat nested interfaces in order to preserve the ability to build kinematic and static admissible fields. This property would be much more difficult to conserve with mortar methods [4, 3]. The algorithm is presented in Subsection 3.3; we choose to solve simultaneously the forward and the adjoint problems on the locally refined mesh, since there is no real-extra cost in solving both problems at the same time. In Subsection 3.4, we verify that the chosen strategy enables us to improve the bound on the estimation.

### 3.2 Interpolation of displacements and forces

We assume that after a first resolution, some subdomains have been locally refined (either in  $h$  or  $p$  sense) or enriched, so that new degrees of freedom have been inserted on the boundary of the domain whereas former boundary nodes have been preserved. Figure 3 presents a  $H/2$  refinement in the case of 2 subdomains. In order to keep the ability of reconstructing globally admissible fields, we decide the (coarse) interface to be the master when imposing displacements or forces on subdomains, and we keep the assembling operators unchanged. We thus only need to derive the interpolation matrix  $\mathbf{N}^{(s)}$  which defines local (fine) displacements  $\mathbf{u}_F^{(s)}$  from interface (coarse) displacement  $\mathbf{u}_C^{(s)} = \mathbf{A}^{(s)T} \mathbf{U}$ :  $\mathbf{u}_F^{(s)} = \mathbf{N}^{(s)} \mathbf{u}_C^{(s)}$ ; and the interpolation matrix  $\mathbf{M}^{(s)}$  which defines local (fine) nodal forces  $\boldsymbol{\lambda}_F^{(s)}$  from interface (coarse) forces  $\boldsymbol{\lambda}_C^{(s)} = \mathbf{B}^{(s)T} \boldsymbol{\Lambda}$ :  $\boldsymbol{\lambda}_F^{(s)} = \mathbf{M}^{(s)} \boldsymbol{\lambda}_C^{(s)}$ .

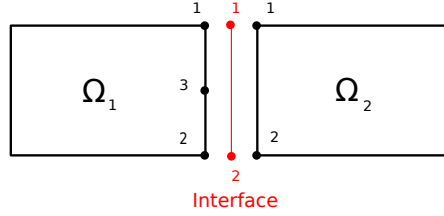


Figure 3: Titre

Let  $\underline{\varphi}_C^{(s)}$  and  $\underline{\varphi}_F^{(s)}$  be the matrices of the coarse and fine shape functions, so that displacements write  $\underline{u}_C^{(s)} = \underline{\varphi}_C^{(s)} \mathbf{u}_C^{(s)}$  on the interface and  $\underline{u}_F^{(s)} = \underline{\varphi}_F^{(s)} \mathbf{u}_F^{(s)}$  on the boundary. Let  $\mathbf{X}_F^{(s)}$  be the coordinates of the fine nodes, we have  $\underline{\varphi}_F^{(s)}(\mathbf{X}_F^{(s)}) = \mathbf{I}$ . Imposing the continuity of the displacement writes:

$$\begin{aligned} \underline{u}_C^{(s)} &= \underline{\varphi}_C^{(s)} \mathbf{u}_C^{(s)} = \underline{\varphi}_F^{(s)} \mathbf{u}_F^{(s)} = \underline{u}_F^{(s)} \\ \Leftrightarrow \underline{\varphi}_C^{(s)}(\mathbf{X}_F^{(s)}) \mathbf{u}_C^{(s)} &= \mathbf{u}_F^{(s)}, \end{aligned}$$

which defines the interpolation matrix for displacement:

$$\mathbf{N}^{(s)} = \underline{\varphi}_C^{(s)}(\mathbf{X}_F^{(s)}). \quad (32)$$

Matrix  $\mathbf{N}^{(s)}$  simply imposes extra fine nodes to follow the displacement imposed by the coarse discretization.

The generation of statically admissible fields requires an algorithm, noted  $\mathcal{G}$  in [28], which defines a continuous distribution of traction  $\underline{t}$  from nodal reactions:  $\underline{t} = \mathcal{G} \boldsymbol{\lambda}$ . Then transfer operator  $\mathbf{M}^{(s)}$  must ensure that the same continuous tractions are created from the coarse and the fine information:  $\underline{t}_F^{(s)} = \underline{t}_C^{(s)}$ . In other words  $\mathcal{G}_F \mathbf{M}^{(s)} = \mathcal{G}_C$ .

Algorithm  $\mathcal{G}$  in [28] consists in finding a distribution of traction on the same finite element subspace as the displacement, under the constraint to preserve the mechanical work:

$$\begin{aligned}\underline{t}_C^{(s)} &= \underline{\varphi}_C^{(s)} \left( \mathbf{G}_C^{(s)} \right)^{-1} \lambda_C^{(s)} \quad \text{with } \mathbf{G}_C^{(s)} = \int_{\Gamma} \underline{\varphi}_C^{(s)T} \underline{\varphi}_C^{(s)} d\Gamma \\ \underline{t}_F^{(s)} &= \underline{\varphi}_F^{(s)} \left( \mathbf{G}_F^{(s)} \right)^{-1} \lambda_F^{(s)} \quad \text{with } \mathbf{G}_F^{(s)} = \int_{\Gamma} \underline{\varphi}_F^{(s)T} \underline{\varphi}_F^{(s)} d\Gamma\end{aligned}\tag{33}$$

$\mathbf{G}_C^{(s)}$  and  $\mathbf{G}_F^{(s)}$  are the  $L^2(\Gamma)$  mass matrices of the coarse and fine interpolations. Because of the chosen interpolation, it is sufficient to satisfy the relationship  $\underline{t}_F^{(s)} = \underline{t}_C^{(s)}$  on the fine nodes, which leads to:

$$\mathbf{M}^{(s)} = \mathbf{G}_F^{(s)} \mathbf{N}^{(s)} (\mathbf{G}_C^{(s)})^{-1}\tag{34}$$

*Remark.* We also have the following classical expressions for matrices  $\mathbf{N}^{(s)}$  and  $\mathbf{M}^{(s)}$ :

$$\begin{aligned}\mathbf{N}^{(s)} &= \left( \mathbf{G}_F^{(s)} \right)^{-1} \mathbf{G}_{FC}^{(s)} \\ \mathbf{M}^{(s)} &= \mathbf{G}_{FC}^{(s)} \left( \mathbf{G}_C^{(s)} \right)^{-1}\end{aligned}\quad \text{with } \mathbf{G}_{FC}^{(s)} = \int_{\Gamma} \underline{\varphi}_F^{(s)T} \underline{\varphi}_C^{(s)} d\Gamma\tag{35}$$

When testing the balance of tractions in the interface displacements we obtain the classical preservation of the mechanical work which writes:

$$\mathbf{N}^{(s)T} \mathbf{M}^{(s)} = \mathbf{I}\tag{36}$$

*Remark.* Of course, for unchanged (non-refined) subdomains, we have  $\mathbf{N}^{(s)} = \mathbf{M}^{(s)} = \mathbf{I}$ .

### 3.3 FETI and BDD algorithms with nonconforming interfaces

The standard BDD algorithm (see for instance Algorithm 1 in [34]) is barely modified, only the assembling operators change depending if they manipulate displacement or forces:

- BDD assembling operator becomes  $\mathbf{A}^{(s)} \mathbf{N}^{(s)T}$ ,
- BDD scaled assembling operator becomes  $\tilde{\mathbf{A}}^{(s)} \mathbf{M}^{(s)T}$ ,

The final algorithm is summarized in Algorithm 2.

The preservation of work (36) together with the definition of the scaling operators (13) ensure that the reconstructed fields satisfy the required continuity or



---

**Algorithm 2:** Block-BDD with incompatibility, unknown  $\mathbf{U} = [\mathbf{u}, \tilde{\mathbf{u}}]$ 


---

```

 $\mathbf{U} = \text{Initialize}(\mathbf{F}^{(s)})$  ;
 $(\mathbf{\Lambda}_D^{(s)}, \mathbf{U}_D^{(s)}) = \text{Solve}_D(\mathbf{N}^{(s)} \mathbf{A}^{(s)T} \mathbf{U}, \mathbf{F}^{(s)})$ ;
Compute residual  $\mathbf{R} = \sum_s \mathbf{A}^{(s)} \mathbf{N}^{(s)T} \mathbf{\Lambda}_D^{(s)}$ ;
Define local traction  $\delta \mathbf{\Lambda}^{(s)} = \mathbf{M}^{(s)} \tilde{\mathbf{A}}^{(s)T} \mathbf{R}$  ; //  $\mathbf{\Lambda}_N^{(s)} = \mathbf{\Lambda}_D^{(s)} - \delta \mathbf{\Lambda}^{(s)}$ 
 $\delta \mathbf{u}^{(s)} = \text{Solve}_N(\delta \mathbf{\Lambda}^{(s)}, 0)$  ; //  $\mathbf{U}_N^{(s)} = \mathbf{U}_D^{(s)} - \delta \mathbf{U}^{(s)}$ 
Preconditioned residual  $\mathbf{Z} = \sum_s \tilde{\mathbf{A}}^{(s)} \mathbf{M}^{(s)T} \delta \mathbf{U}^{(s)}$  ;
Search direction  $\mathbf{W} = \mathbf{P}_1 \mathbf{Z}$ ;
while  $\sqrt{\mathbf{R}^T \mathbf{Z}} > \epsilon$  do
     $(\delta \mathbf{\Lambda}_D^{(s)}, \delta \mathbf{U}_D^{(s)}) = \text{Solve}_D(\mathbf{N}^{(s)} \mathbf{A}^{(s)T} \mathbf{W}, 0)$ ;
     $\mathbf{Q} = \sum_s \mathbf{A}^{(s)} \mathbf{N}^{(s)T} \delta \mathbf{\Lambda}_D^{(s)}$ ;
     $\alpha = (\mathbf{Q}^T \mathbf{W})^{-1} (\mathbf{R}^T \mathbf{Z})$ ;
     $\mathbf{U} \leftarrow \mathbf{U} + \mathbf{W} \alpha$  ; //  $\mathbf{U}_D^{(s)} \leftarrow \mathbf{U}_D^{(s)} + \delta \mathbf{U}_D^{(s)} \alpha$ 
    //  $\mathbf{\Lambda}_D^{(s)} \leftarrow \mathbf{\Lambda}_D^{(s)} + \delta \mathbf{\Lambda}_D^{(s)} \alpha$ 
     $\mathbf{R} \leftarrow \mathbf{R} - \mathbf{Q} \alpha$ ;
     $\delta \mathbf{\Lambda}^{(s)} = \mathbf{M}^{(s)} \tilde{\mathbf{A}}^{(s)T} \mathbf{R}$  ; //  $\mathbf{\Lambda}_N^{(s)} = \mathbf{\Lambda}_D^{(s)} - \delta \mathbf{\Lambda}^{(s)}$ 
     $\delta \mathbf{U}^{(s)} = \text{Solve}_N(\delta \mathbf{\Lambda}^{(s)}, 0)$  ; //  $\mathbf{U}_N^{(s)} = \mathbf{U}_D^{(s)} - \delta \mathbf{U}^{(s)}$ 
     $\mathbf{Z} = \sum_s \tilde{\mathbf{A}}^{(s)} \mathbf{M}^{(s)T} \delta \mathbf{U}_b^{(s)}$ ;
     $\mathbf{W} \leftarrow \mathbf{P}_1 \mathbf{Z} - \mathbf{W} (\mathbf{Q}^T \mathbf{W})^{-1} (\mathbf{Q}^T \mathbf{P}_1 \mathbf{Z})$ 
end

```

---

balance on the interface. More precisely in the BDD algorithm, the displacement is built as a continuous field, whereas the nodal forces  $\mathbf{\Lambda}_N^{(s)}$  are balanced on the interface because:

$$\begin{aligned}
 \sum_s \mathbf{A}^{(s)} \mathbf{N}^{(s)T} \mathbf{\Lambda}_N^{(s)} &= \sum_s \mathbf{A}^{(s)} \mathbf{N}^{(s)T} (\mathbf{\Lambda}_D^{(s)} - \delta \mathbf{\Lambda}^{(s)}) \\
 &= (\mathbf{I} - \sum_s \mathbf{A}^{(s)} \mathbf{N}^{(s)T} \mathbf{M}^{(s)} \tilde{\mathbf{A}}^{(s)T}) \mathbf{R} \\
 &= (\mathbf{I} - \sum_s \mathbf{A}^{(s)} \tilde{\mathbf{A}}^{(s)T}) \mathbf{R} = 0
 \end{aligned} \tag{37}$$

*Remark.* Similarly, FETI algorithm can be derived from Algorithm 2 in [34] by simply replacing the assembling operators:

- FETI assembly operator becomes  $\mathbf{B}^{(s)} \mathbf{M}^{(s)T}$ ,
- FETI scaled assembling operator becomes  $\tilde{\mathbf{B}}^{(s)} \mathbf{N}^{(s)T}$ .

*Remark.* When the solver is converged, because of the incompatible meshes, the displacement fields  $(\underline{u}_D^{(s)})_s$  and  $(\underline{u}_N^{(s)})_s$  are no longer identical.

### 3.4 Assessment

Let us reuse the example of previous section. Figure (4a) plots the error map of the adjoint problem on a first coarse mesh, it is essentially localized within the central subdomain; where the loading of the adjoint problem is defined.

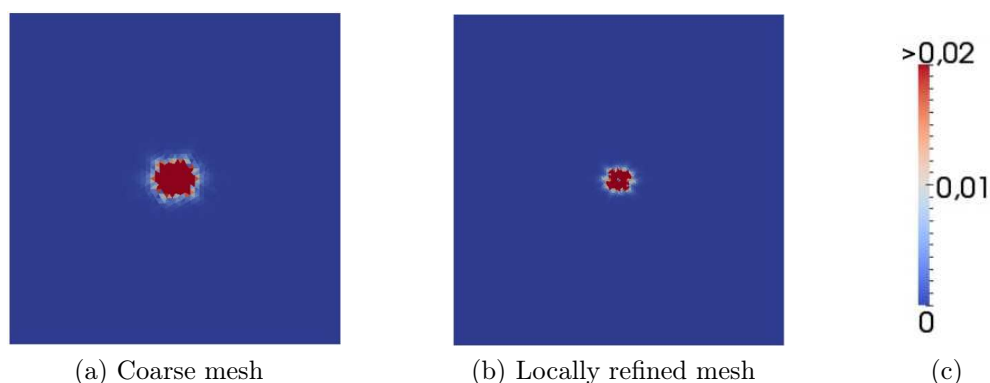


Figure 4: Effect of local refinement (map of the error estimator) on the adjoint problem

Accordingly, from the point of view of the adjoint (resp. forward) problem, we need to refine the mesh of the central subdomain. Three cases are considered: two with conforming meshes (i) a coarse mesh (given  $H$ ) (ii) a fine mesh ( $h = H/4$ ), and one with non-conforming meshes due local refinement only in the central subdomain ( $h = H/4$ , see Figure 5 for a schematic presentation with a coarser mesh than in actual computations). In all cases, we compute the error estimator at convergence of the solver. The number of Conjugate Gradient iterations is identical for the 3 meshes (9 iterations). The results are summarized in Table 2. When the central subdomain is refined, the error estimator for the adjoint problem decreases while the error estimator of the forward problem remains more or less unchanged. As a consequence, the use of local refinement (independently by subdomains) enables us to obtain improved bounds on the quantity of interest ( $-32\%$ ) with a simple implementation and less degrees of freedom ( $+38\%$  #dof) than would be required by the full fine mesh ( $+300\%$  #dof for a  $-65\%$  decrease of the estimator).

Figure 4b plots the map of error for the adjoint problem when only the central subdomain is refined. The source of error due to the non-conforming interface (boundary of the central subdomain) is clearly negligible.

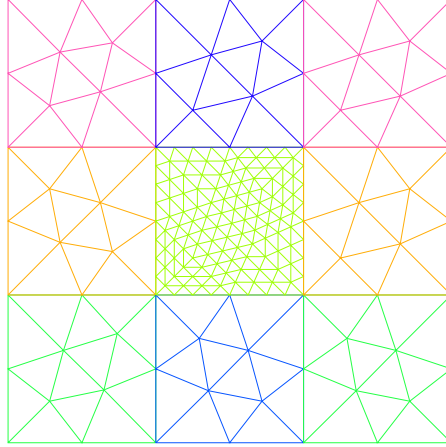


Figure 5: Hierarchically refined mesh on the central subdomain

case	$e$	$\tilde{e}$	$I_H$	$I_{HH,2}$	$\frac{1}{2}e\tilde{e}$	#dof
Coarse mesh	2.418	10.23	2.916	$3.8575 \cdot 10^{-4}$	12.37	8738
Fine mesh	1.216	7.020	2.916	$-9.695 \cdot 10^{-4}$	4.268	34988
Refined central SD	2.404	7.020	2.9169	$3.8575 \cdot 10^{-4}$	8.438	12122

Table 2: Comparaison between conforming and non conforming cases,  $I_{ex} = 2,916$

Of course, the use of locally refined meshes is questionable if the zone of high contribution to the error is close to a non-matching interface: in that case the driving of the fine kinematic by the coarse interface is an extra source of error. To evaluate that problem, we study how the efficiency of the local refinement is altered by the distance  $d$  between the support of the quantity of interest  $\omega = [-\frac{L}{10} - d; \frac{L}{10} - d] \times [-\frac{L}{10}; \frac{L}{10}]$  and the interface  $\Gamma$ .

For each configuration (characterized by the distance  $d$ ) we compute the relative discretization error (discretization error divided by the energy of the finite element solution) of the adjoint problem  $\rho_{Hh}$  (resp.  $\rho_h$ ) when  $h = H/4$  only in the central subdomain (resp. on all subdomains). Figure 6 plots the ratio  $\rho_{Hh}/\rho_h$  as a function of the distance  $d/d_0$  (where  $d = d_0$  corresponds to the quantity of interest at the center of the subdomain). The quality of the refined estimator is degraded only when  $d/d_0 < 10^{-1}$  which corresponds to a quantity of interest very close to the interface. Thus the idea to conduct local refinements remains valid in a wide range of configurations.

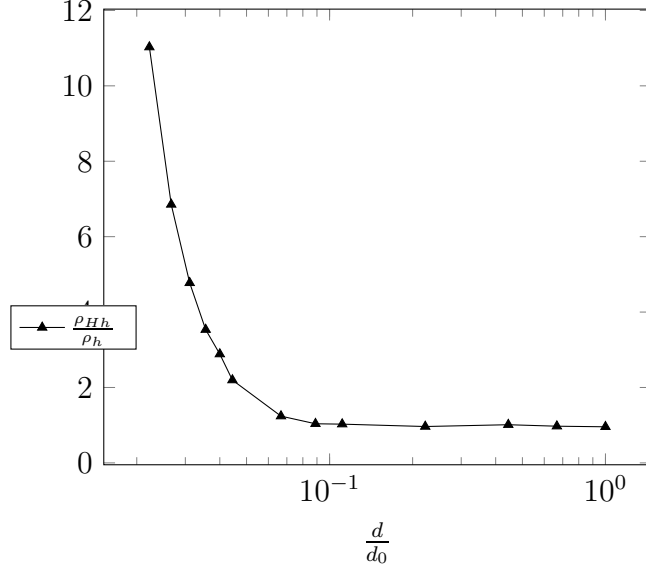


Figure 6: Influence of the distance  $d$

## 4 Numerical experiments

The aim of these assessments is to show how easily the error bound can be improved by the refinement strategy described in Section 3.

We always use the stopping criterion proposed in Section 2: iterations are stopped when the algebraic error becomes negligible (10 times smaller) with respect to the contribution of the discretization estimated at the first iteration, for both forward and adjoint problems. We verify that the discretization error computed with the new interface fields barely varies, as could be expected from [34].

After a first resolution on a coarse mesh, we analyze the contributions of sub-domains to the forward and adjoint error estimator using the following measures:

$$\eta^{(s)} = \frac{e_{CR_{\Omega(s)}}(\underline{u}_N^{(s)}, \hat{\underline{\sigma}}_N^{(s)})}{\sqrt{\sum_s e_{CR_{\Omega(s)}}^2(\underline{u}_N^{(s)}, \hat{\underline{\sigma}}_N^{(s)})}} \quad \text{and} \quad \tilde{\eta}^{(s)} = \frac{e_{CR_{\Omega(s)}}(\tilde{\underline{u}}_N^{(s)}, \hat{\underline{\sigma}}_N^{(s)})}{\sqrt{\sum_s e_{CR_{\Omega(s)}}^2(\tilde{\underline{u}}_N^{(s)}, \hat{\underline{\sigma}}_N^{(s)})}} \quad (38)$$

From that information we deduce h-refinement strategies and evaluate their ability to improve the estimator.

### 4.1 The Gamma-shaped structure

We consider a Gamma-shaped structure clamped on its base and submitted to prescribed traction and shear on the upper-right side. The problem is linear elastic

with unit Young modulus and 0.3 Poisson coefficient. The initial mesh is made out of first order triangles. The structure is decomposed into 7 domains and we use the classical BDD solver. The quantity of interest is the mean value of the strain  $\varepsilon_{yy}$  over the region  $\omega$  located in the seventh subdomain, as illustrated in Figure 7. We build statically admissible stress fields with the Flux-free technique [7, 27]. Each star-patch problem is solved with a forth order interpolation.

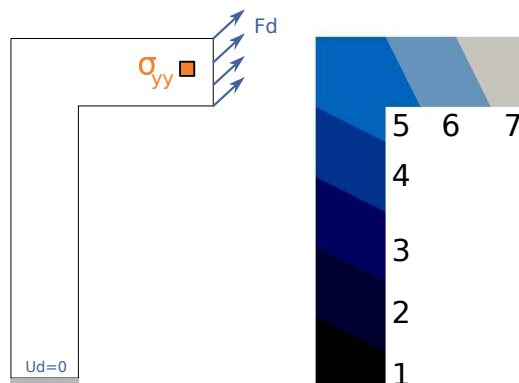


Figure 7: Loading of forward (blue) and adjoint problems (orange)

The contributions  $\eta^{(s)}$  and  $\tilde{\eta}^{(s)}$  to the error estimator on the initial mesh are presented in Figure 8a. We observe that the major contribution to the error for the forward problem is located in the fifth subdomain (because of the corner). The objective is to refine this domain in order to obtain a level of error comparable to subdomain 1 (which is the second largest contributor). It implies to divide the error in the fifth subdomain by 2.8. The convergence in this subdomain is driven by the corner singularity. Therefore, we decide to perform three hierarchical h-refinements in the fifth subdomain. The quantity of interest being localized in Subdomain 7, we also process 3 hierarchical h-refinements in that subdomain.

As expected, we can see on Figure 8b that the forward error in Subdomain 5 and Subdomain 1 are nearly the same on the new mesh. We also observe that the error of the adjoint problem has decreased (due to the refinement of Subdomain 7). In Table 3, we sum up the global values of discretization errors, the quantities  $I_H$  and  $I_{HH,2}$ .

Mesh	$e$	$\tilde{e}$	$I_H$	$I_{HH,2}$	$\frac{1}{2}e\tilde{e}$
Uniform	33.528	0.20977	-0.39157	-0.0010045	3.5165
Locally refined	18.464	0.072916	-0.39006	-0.00050892	0.67316

Table 3: Performance of local refinement on the  $\Gamma$ -shaped structure

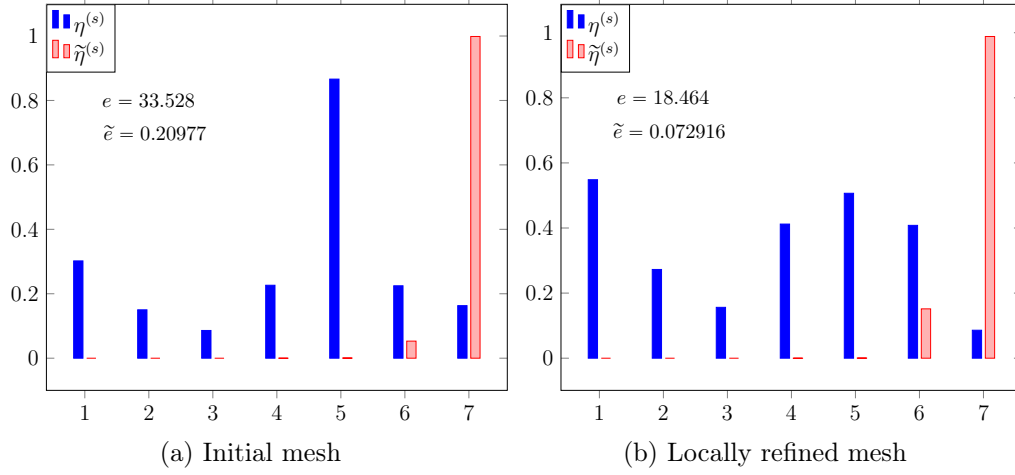


Figure 8: Distribution of the error estimator within subdomains

## 4.2 The pre-cracked structure

We now consider the pre-cracked structure decomposed into 16 subdomains as illustrated in Figure 9. The displacements at the base of the structure and on the bigger hole are imposed to be zero. The upper-left part and the second hole are subjected to a constant unit pressure. The quantity of interest is the mean of the stress component  $\sigma_{xx}$  on a region  $\omega$  close to the crack. In Figure 9, the loading of the reference problem is in blue and the loading of adjoint problem is in orange. We used the FETI algorithm (dual approach) to solve the interface problem and the statically admissible stress fields are built using the Flux-free technique [7, 27] with forth order interpolation for problems on star-patches.

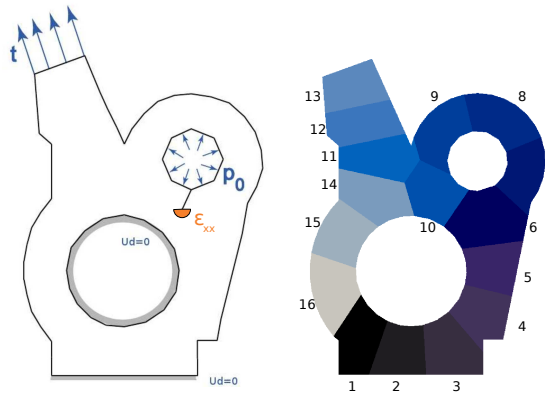


Figure 9: Loading of forward (blue) and adjoint problems (orange), domain decomposition

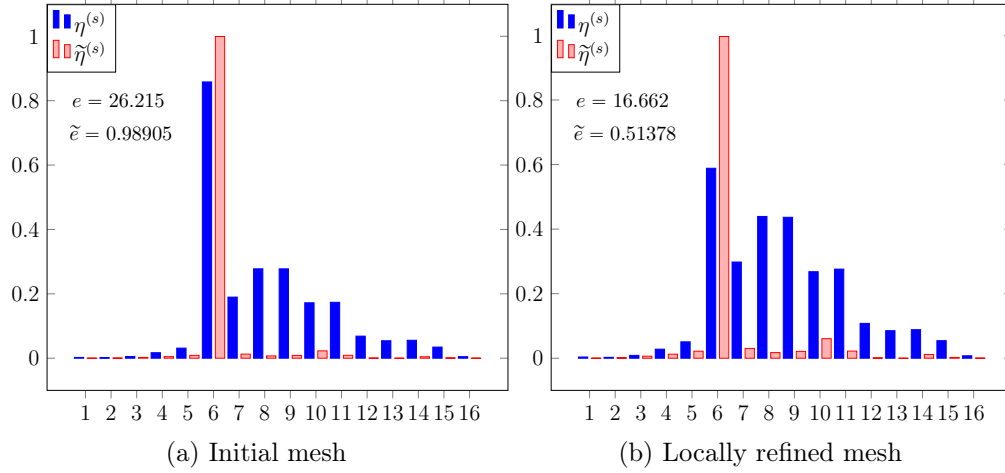


Figure 10: Distribution of error within subdomains

In Figure 10a, we observe that the discretization errors of the forward and adjoint problem are quasi totally located in the sixth subdomain. Therefore, we perform hierarchical refinement in this subdomain in order to reach a level of forward error comparable to the second larger contributor (Subdomain 8). That corresponds to dividing the error in Subdomain 6 by a factor 3. Because of the singularity in the subdomain, convergence is expected to evolve like  $\sqrt{h}$  so we perform 4 hierarchical refinements (each coarse element edge is divided into 16 fine edges).

As can be seen in Figure 10b, after local refinement the contributions of Subdomains 6 and 8 to the forward error are equivalent, and the adjoint error also decreased, as expected. The global performance is summed up in Table 4. In the end, the bound on the estimator on the quantity of interest was decreased by a factor 3.

Mesh	$e$	$\tilde{e}$	$I_H$	$I_{HH,2}$	$\frac{1}{2}e\tilde{e}$
Uniform	26.215	0.98905	2.4915	3.1935	12.964
Locally refined	16.662	0.51378	3.2165	0.086055	4.2803

Table 4: Performance of local refined for the cracked structure

## 5 Conclusion

In this article, we extended two classical results of goal-oriented error estimation to problems solved by domain decomposition methods. We solved simultaneously

both forward and adjoint problems on the same mesh using a block conjugate gradient algorithm and proposed a fully parallel procedure to recover admissible fields and estimate the error on the quantity of interest. Moreover, the error bounds separated the contribution of the discretization from the contribution of the iterative solver which lead to defining of a stopping criterion of the iterative solver that prevented oversolving.

We also showed that the method could be easily extended to the case of locally enriched kinematics (using  $h$  or  $p$  refinements or dedicated enrichments) only constrained to contain the original coarse kinematic on the boundary of subdomains. This property provided a simple parallel way to improve estimators as soon as the singularities (local quantity of interests, crack tips, etc) were not too close to the interface. The concepts were validated on academic examples using the most basic  $h$ -refinement strategy. Of course using ad-hoc enrichment instead of brutal  $h$ -refinement should improve the performance of the method.

Future work will focus on exploiting the fact that the shape of the interface is preserved by the enrichment to define acceleration strategies, and on defining original strategies to balance the computational load between subdomains for the resolution of the refined problem.

## References

- [1] I. Babuška and W. C. Rheinboldt. Computational error estimates and adaptive processes for some nonlinear structural problems. *Computer Methods in Applied Mechanics and Engineering*, 34(1-3):895–937, 1982.
- [2] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, A. Iserles (ed.), Cambridge University Press, 10:1–120, 2001.
- [3] Fakher Ben Belgacem. The mortar finite element method with lagrange multipliers. *Numerische Mathematik*, 84(2):173–197, 1999.
- [4] C. Bernardi, Y. Maday, and A. T. Patera. A new nonconforming approach to domain decomposition: the mortar element method. In H. Brezis & J.-L. Lions, editor, *Collège de France Seminar XI*, volume XI, page 13. Pitman, 1994.
- [5] L. Chamoin and P. Ladevèze. A non-intrusive method for the calculation of strict and efficient bounds of calculated outputs of interest in linear viscoelasticity problems. *Computer Methods in Applied Mechanics and Engineering*, 197(9-12):994–1014, 2008.



- [6] L. Chamoin and P. Ladevèze. Strict and practical bounds through a non-intrusive and goal-oriented error estimation method for linear viscoelasticity problems. *Finite Elements in Analysis and Design*, 45(4):251–262, 2009.
- [7] R. Cottureau, P. Díez, and A. Huerta. Strict error bounds for linear solid mechanics problems using a subdomain-based flux-free method. *Computational Mechanics*, 44(4):533–547, 2009.
- [8] T. Coupez, H. Dignonnet, and R. Ducloux. Parallel meshing and remeshing. *Applied mathematical modelling*, 25(2):153–175, 2000.
- [9] C. Farhat and F. X. Roux. The dual schur complement method with well-posed local neumann problems. *Contemporary Mathematics*, 157:193, 1994.
- [10] Pierre Gosselet and Christian Rey. Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering*, 13(4):515–572, 2006.
- [11] T. Grätsch and F. Hartmann. Pointwise error estimation and adaptivity for the finite element method using fundamental solutions. *Computational Mechanics*, 37:394–407, 2006. 10.1007/s00466-005-0711-4.
- [12] Axel Klawonn and Olof Widlund. Feti and neumann-neumann iterative substructuring methods: Connections and new results. *Communications on Pure and Applied Mathematics*, 54(1):57–90, 2001.
- [13] P. Ladevèze. Constitutive relation errors for f.e. analysis considering (visco-) plasticity and damage. *International Journal for Numerical Methods in Engineering*, 52(56):527–542, 2001.
- [14] P. Ladevèze. Upper error bounds on calculated outputs of interest for linear and nonlinear structural problems. *Comptes Rendus Académie des Sciences - Mécanique, Paris*, 334(7):399–407, 2006.
- [15] P. Ladevèze. Strict upper error bounds on computed outputs of interest in computational structural mechanics. *Computational Mechanics*, 42(2):271–286, 2008.
- [16] P. Ladevèze, G. Coffignal, and J. P. Pelle. Accuracy of elastoplastic and dynamic analysis. *Accuracy Estimates and Adaptive Refinements in Finite Element Computations (Babuška, Gago, Oliveira and Zienkiewicz Editors)*, Wiley, London, 11:181–203, 1986.

- [17] P. Ladevèze, J. P. Pelle, and P. Rougeot. Error estimation and mesh optimization for classical finite elements. *Engineering Computations*, 8(1):69, 1991.
- [18] P. Ladevèze and P. Rougeot. New advances on a posteriori error on constitutive relation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 150(1-4):239–249, 1997.
- [19] Patrick Le Tallec. Domain decomposition methods in computational mechanics. *Comput. Mech. Adv.*, 1(2):121–220, 1994.
- [20] F. Louf, J. P. Combe, and J. P. Pelle. Constitutive error estimator for the control of contact problems involving friction. *Computers and Structures*, 81(18-19):1759–1772, 2003.
- [21] D. Meidner, R. Rannacher, and J. Vihharev. Goal-oriented error control of the iterative solution of finite element equations. *Journal of numerical mathematics*, 17(2):143–172, 2009.
- [22] S. Ohnibus, E. Stein, and E. Walhorn. Local error estimates of fem for displacements and stresses in linear elasticity by solving local neumann problems. *International Journal for Numerical Methods in Engineering*, 52(7):727–746, 2001.
- [23] Dianne P. O’Leary. The block conjugate gradient algorithm and related methods. *Linear Algebra Appl.*, 29:293–322, 1980.
- [24] M. Paraschivoiu. A posteriori finite element output bounds in three space dimensions using the feti method. *Computer Methods in Applied Mechanics and Engineering*, 190(49-50):6629–6640, 2001.
- [25] M. Paraschivoiu and H. W. Choi. A posteriori finite element output bounds with adaptive mesh refinement: application to a heat transfer problem in a three-dimensional rectangular duct. *Computer Methods in Applied Mechanics and Engineering*, 191(43):4905–4925, 2002.
- [26] M. Paraschivoiu, J. Peraire, and A. T. Patera. A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 150(1-4):289–312, 1997.
- [27] N. Parés, P. Díez, and A. Huerta. Subdomain-based flux-free a posteriori error estimators. *Computer Methods in Applied Mechanics and Engineering*, 195(4-6):297–323, 2006.

- [28] A. Parret-Fréaud, C. Rey, P. Gosselet, and F. Feyel. Fast estimation of discretization error for fe problems solved by domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(49-52):3315–3323, 2010.
- [29] A. T. Patera and Einar M. Ronquist. A general output bound result: application to discretization and iteration error estimation and control. *Mathematical Models and Methods in Applied Sciences*, 11(04):685–712, 2001.
- [30] F. Pled, L. Chamoin, and P. Ladevèze. On the techniques for constructing admissible stress fields in model verification: Performances on engineering examples. *International Journal for Numerical Methods in Engineering*, 88(5):409–441, 2011.
- [31] S. Prudhomme and J. T. Oden. On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Computer Methods in Applied Mechanics and Engineering*, 176(1-4):313–331, 1999.
- [32] R Rannacher, A Westenberger, and W Wollner. Adaptive finite element solution of eigenvalue problems: balancing of discretization and iteration error. *Journal of Numerical Mathematics*, 18(4):303–327, 2010.
- [33] V. Rey, P. Gosselet, and C. Rey. Study of the strong prolongation equation for the construction of statically admissible stress fields: Implementation and optimization. *Computer Methods in Applied Mechanics and Engineering*, 268(0):82 – 104, 2014.
- [34] Valentine Rey, Christian Rey, and Pierre Gosselet. A strict error bound with separated contributions of the discretization and of the iterative solver in non-overlapping domain decomposition methods. *Computer Methods in Applied Mechanics and Engineering*, 270(0):293 – 303, 2014.
- [35] Daniel J. Rixen and Charbel Farhat. A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods in Engineering*, 44(4):489–516, 1999.
- [36] Y. Saad. *Iterative methods for sparse linear systems*. PWS Publishing Company, 3rd edition edition, 2000.
- [37] Yousef Saad. Analysis of augmented krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 18(2):435–449, 1997.
- [38] R. Stevenson. The completion of locally refined simplicial partitions created by bisection. *Mathematics of computation*, 77(261):227–241, 2008.

- [39] T. Strouboulis, I. Babuška, D. K. Datta, K. Copps, and S. K. Gangaraj. A posteriori estimation and adaptive control of the error in the quantity of interest. part i: A posteriori estimation of the error in the von mises stress and the stress intensity factor. *Computer Methods in Applied Mechanics and Engineering*, 181(1–3):261–294, 2000.
- [40] S. Tavener and T. Wildey. Adjoint based a posteriori analysis of multiscale mortar discretizations with multinumerics. *SIAM Journal on Scientific Computing*, 35(6):A2621–A2642, 2013.
- [41] Julien Waeytens, Ludovic Chamoin, and Pierre Ladevèze. Guaranteed error bounds on pointwise quantities of interest for transient viscodynamics problems. *Comput. Mech.*, 49(3):291–307, 2012.